

BEST AVAILABLE COPY

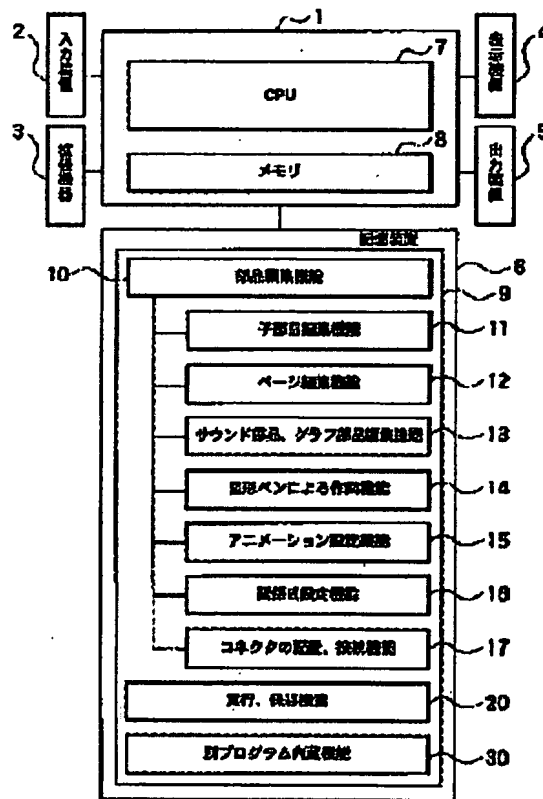
MULTIMEDIA AUTHORIZING TOOL AND RECORDING MEDIUM WITH AUTHORIZING PROGRAM RECORDED ON IT

Patent number: JP2002015333
Publication date: 2002-01-18
Inventor: SAKAMOTO MASARU
Applicant: MICRO BRAIN KK
Classification:
 - international: G06T13/00; G06T11/60; H04N1/387; H04N5/262; H04N5/91
 - european:
Application number: JP20000199412 20000630
Priority number(s): JP20000199412 20000630

Report a data error here

Abstract of JP2002015333

PROBLEM TO BE SOLVED: To provide an authoring program capable of easily originating, executing and distributing move effective digital contents without requiring advanced technical knowledge or complex operations. **SOLUTION:** This recording medium has the authoring program recorded for producing multimedia contents or the like based on digital material data such as movies, still images, voices, figures, letters, signs, sentences, programs, etc. The authoring program has a parts editing function for originating multimedia contents by arranging and editing the digital material data as parts having therein data and relational expressions and an executing function for executing the multimedia contents being originated or already originated.



Data supplied from the esp@cenet database - Worldwide

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-15333

(P2002-15333A)

(43) 公開日 平成14年1月18日 (2002.1.18)

(51) Int.Cl. ⁷	識別記号	FI	特許出願公開番号
G 0 6 T 13/00		G 0 6 T 13/00	B 5 B 0 5 0
11/60	2 0 0	11/60	5 C 0 2 3
H 0 4 N 1/387		H 0 4 N 1/387	5 C 0 5 3
5/262		5/262	5 C 0 7 6
5/91		5/91	N

審査請求 未請求 請求項の数16 OL (全 19 頁)

(21) 出願番号 特願2000-199412(P2000-199412)

(22) 出願日 平成12年6月30日 (2000.6.30)

(71) 出願人 500310959

株式会社マイクロブレイン

東京都稲城市東長沼1066番地4号

(72) 発明者 坂本 勝

東京都稲城市東長沼1066番地4号

(74) 代理人 100083806

弁理士 三好 秀和 (外8名)

Fターム(参考) 5B050 AA06 AA10 BA06 BA08 CA07

EA12 EA19 EA24 FA02 FA10

5C023 AA02 AA03 AA11 BA11 DA01

5C053 FA05 FA14 FA23 JA07 KA05

LA04 LA06 LA14

5C076 AA17 AA21 AA22 AA24 BA08

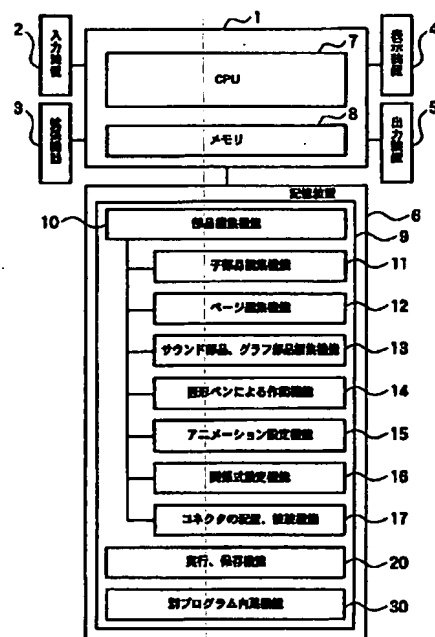
CA02

(54) 【発明の名称】 マルチメディアオーサリングツール及びオーサリングプログラムを記録した記録媒体

(57) 【要約】

【課題】 高度な技術的知識や複雑な操作を必要とせず、より効果的なデジタルコンテンツを簡単に作成、実行、配布することができるオーサリングプログラムを提供すること。

【解決手段】 動画、静止画、音声、図形、文字、記号、文章、プログラム等のデジタル素材データをもとにマルチメディアコンテンツ等を製作するためのオーサリングプログラムを記録した記録媒体であって、前記オーサリングプログラムは、前記デジタル素材データを、内部にデータと関係式とを有する部品として、配置・編集してマルチメディアコンテンツを作成する部品編集機能と、作成中のあるいは作成したマルチメディアコンテンツを実行する実行機能とを有することによる。



【特許請求の範囲】

【請求項1】 動画、静止画、音声、図形、文字、記号、文章、プログラム等のデジタル素材データをもとにマルチメディアコンテンツ等を製作するためのオーサリングプログラムを記録した記録媒体であって、前記オーサリングプログラムは、前記デジタル素材データを、内部にデータと関係式とを有する部品として、配置・編集してマルチメディアコンテンツを作成する部品編集機能と、作成中のあるいは作成したマルチメディアコンテンツを実行する実行機能と、を有することを特徴とするオーサリングプログラムを記録した記録媒体。

【請求項2】 前記部品は、複数個からなる部品の一方を親部品、他方を子部品として組み合わせた階層構造からなり、前記部品編集機能は、前記子部品が前記親部品の表示座標系を継承し、前記親部品の縮小・拡大・回転等の変形動作に合わせて前記子部品も変形動作するように座標計算することを特徴とする請求項1に記載のオーサリングプログラムを記録した記録媒体。

【請求項3】 前記座標計算は、前記階層構造の階層分だけ座標変換を繰り返して求めた座標を用いて写像のための一次式を作成し、ガウスの除去法により前記一次式の係数を求め、求めた係数によって前記子部品の表示画像を描画することを特徴とする請求項2に記載のオーサリングプログラムを記録した記録媒体。

【請求項4】 前記部品編集機能は、前記親部品の変形動作による子部品の変形を打ち消すロック機能を有することを特徴とする請求項3に記載のオーサリングプログラムを記録した記録媒体。

【請求項5】 前記部品は、部品内部における並列構造として複数のページを有し、前記部品編集機能は、前記ページ間の状態を線形補間することで、中間の状態を表現することを特徴とする請求項1ないし請求項4のいずれかに記載のオーサリングプログラムを記録した記録媒体。

【請求項6】 前記部品がサウンド部品である場合、このサウンド部品のページ内における配置位置座標を、当該サウンド部品を再生する際の左右方向の音量バランスとすることを特徴とする請求項1ないし請求項5のいずれかに記載のオーサリングプログラムを記録した記録媒体。

【請求項7】 前記部品編集機能は、図形部品を作成するための作図部品を有し、前記作図部品の内部に任意の画像データの子部品として配置することで、前記作図部品で作成された前記図形部品は前記画像データによって描画されることを特徴とする請求項2ないし請求項6のいずれかに記載のオーサリングプログラムを記録した記録媒体。

【請求項8】 前記作図部品は、線によって図形部品を

作成するための作図部品であり、前記線が前記画像データによって描画されることを特徴とする請求項7に記載のオーサリングプログラムを記録した記録媒体。

【請求項9】 前記作図部品は、面によって図形部品を作成するための作図部品であり、前記線が前記画像データによって描画されることを特徴とする請求項7に記載のオーサリングプログラムを記録した記録媒体。

【請求項10】 前記実行機能は、前記子部品に対する移動や回転等の操作がなされた場合に、対応する親部品のページを前記子部品の移動量や回転角度に応じて選移させることを特徴とする請求項5ないし請求項9のいずれかに記載のオーサリングプログラムを記録した記録媒体。

【請求項11】 前記部品編集機能は、前記関係式の構文を解析して当該関係式に係る変数の参照関係に関する参照テーブルを作成し、

前記実行機能は、コンテンツを実行する際に、前記変数の値に変化が生じたのを契機に、前記参照テーブルの参照関係から当該変数を参照している関係式を特定し、特定した当該関係式を実行することを特徴とする請求項1ないし請求項10のいずれかに記載のオーサリングプログラムを記録した記録媒体。

【請求項12】 前記関係式の一部あるいは全部が方程式によって記述されていることを特徴とする請求項11に記載のオーサリングプログラムを記録した記録媒体。

【請求項13】 前記関係式が条件文を含む式である場合、当該条件を構成する変数の値に変化が生じたのを契機に当該条件が評価され、その評価に応じた分岐先の式を実行することを特徴とする請求項11ないし請求項12のいずれかに記載のオーサリングプログラムを記録した記録媒体。

【請求項14】 前記部品編集機能は、前記部品間を接続するためのコネクタを有し、前記コネクタは、接続した部品が有する関係式に係る変数同士を論理的に結合し、その値の受け渡しを行うことを特徴とする請求項1ないし請求項13のいずれかに記載のオーサリングプログラムを記録した記録媒体。

【請求項15】 接続した部品は、外部プログラムによって作成された部品であることを特徴とする請求項14に記載のオーサリングプログラムを記録した記録媒体。

【請求項16】 動画、静止画、音声、図形、文字、記号、文章、プログラム等のデジタル素材データ入力する入力装置と、

記憶装置からローディングしたプログラムの処理手続きに従って、入力したデジタル素材データをもとにデジタルコンテンツを作成するCPUと、

CPUにて作成中あるいは作成したデジタルコンテンツを表示する表示装置と、

請求項1ないし請求項15のいずれかに記載のオーサリングプログラムを記録した記録媒体を格納する記憶装置

と、を有し、
前記デジタル素材データを、内部にデータと関係式とを有する部品として、配置・編集してマルチメディアコンテンツを作成することを特徴とするオーサリングツール。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、マルチメディアコンテンツ等をデザイン・編集するためのマルチメディアオーサリングツール及びオーサリングプログラムを記録した記録媒体に関する。

【0002】

【従来の技術】文字、静止画、音声、動画などのデジタル素材を組み合わせて、グラフィック、サウンド、アニメーション、テキストなどのマルチメディアコンテンツをデザイン・編集するためのマルチメディアオーサリングツールが注目されている。このようなオーサリングツールを用いれば、例えばCやC++などのプログラミング言語やOS等に関する高度な知識を必要とせずに、比較的簡単にマルチメディアコンテンツをデザイン・編集することができることから、パーソナルコンピュータの普及とともに、一般ユーザーの間でも使われるようになってきた。

【0003】作成されたマルチメディアコンテンツは、例えば、研究発表や製品デモなどのプレゼンテーションや、教育用ソフト、エデュテインメント・ソフト (Edutainment Software)、ゲーム、百科事典、など、幅広い分野に渡って利用されている。

【0004】

【発明が解決しようとする課題】ところで、従来のオーサリングツールでは、より効果的で高度な編集機能を用いたデジタルコンテンツ、例えば、アニメーションなどを作成しようとする、出力メディアやターゲット環境の特性や機能の限界を考慮に入れた制作技術が必要となったり、複雑な編集操作を要することもあるため、特に一般ユーザーにとって大きなハードルとなっている。

【0005】更にまた、より効果的なデジタルコンテンツを作成しようとする、出来上がりサイズが大きなものになるため、それに見合った記憶容量が必要となったり、複雑な処理を行うことから高性能のCPUが必要になったりする。こうしたデジタルコンテンツの制作環境やターゲット環境は、今後、パーソナルコンピュータのみならず、PDA (Personal Digital Assistants) や携帯電話、更には情報家電へと広がっていくことが考えられることから、コンパクトで軽快なオーサリングツールやデジタルコンテンツが求められる。

【0006】本発明は、このような問題点を鑑みなされたものであって、高度な技術的知識や複雑な操作を必要とせずに、より効果的なデジタルコンテンツを簡単に作成、実行、配布することができるマルチメディアオーサ

リングツール及びオーサリングプログラムを記録した記録媒体を提供することを目的とする。

【0007】

【課題を解決するための手段】上記課題を解決するために、本発明は、動画、静止画、音声、図形、文字、記号、文章、プログラム等のデジタル素材データをもとにマルチメディアコンテンツ等を製作するためのオーサリングプログラムを記録した記録媒体であって、前記オーサリングプログラムは、前記デジタル素材データを、内部にデータと関係式とを有する部品として、配置・編集してマルチメディアコンテンツを作成する部品編集機能と、作成中のあるいは作成したマルチメディアコンテンツを実行する実行機能と、を有することを特徴とする。

【0008】また、前記部品は、複数個からなる部品の一方を親部品、他方を子部品として組み合わせた階層構造からなり、前記部品編集機能は、前記子部品が前記親部品の表示座標系を継承し、前記親部品の縮小・拡大・回転等の変形動作に合わせて前記子部品も変形動作するように座標計算することを特徴とする。

【0009】また、前記座標計算は、前記階層構造の階層分だけ座標変換を繰り返して求めた座標を用いて写像のための一次式を作成し、ガウスの除去法により前記一次式の係数を求め、求めた係数によって前記子部品の表示画像を描画することを特徴とする。

【0010】また、前記部品編集機能は、前記親部品の変形動作による子部品の変形を打ち消すロック機能を有することを特徴とする。

【0011】また、前記部品は、部品内部における並列構造として複数のページを有し、前記部品編集機能は、前記ページ間の状態を線形補間することで、中間の状態を表現することを特徴とする。

【0012】また、前記部品がサウンド部品である場合、このサウンド部品のページ内における配置位置座標を、当該サウンド部品を再生する際の左右方向の音量バランスとすることを特徴とする。

【0013】また、前記部品編集機能は、図形部品を作成するための作図部品を有し、前記作図部品の内部に任意の画像データを子部品として配置することで、前記作図部品で作成された前記図形部品は前記画像データによって描画されることを特徴とする。

【0014】また、前記作図部品は、線によって図形部品を作成するための作図部品であり、前記線が前記画像データによって描画されることを特徴とする。

【0015】また、前記作図部品は、面によって図形部品を作成するための作図部品であり、前記線が前記画像データによって描画されることを特徴とする。

【0016】また、前記実行機能は、前記子部品に対する移動や回転等の操作がなされた場合に、対応する親部品のページを前記子部品の移動量や回転角度に応じて選移させることを特徴とする。

【0017】また、前記部品編集機能は、前記関係式の構文を解析して当該関係式に係る変数の参照関係に関する参照テーブルを作成し、前記実行機能は、コンテンツを実行する際に、前記変数の値に変化が生じたのを契機に、前記参照テーブルの参照関係から当該変数を参照している関係式を特定し、特定した当該関係式を実行することを特徴とする。

【0018】また、前記関係式の一部あるいは全部が方程式によって記述されていることを特徴とする。

【0019】また、前記関係式が条件文を含む式である場合、当該条件を構成する変数の値に変化が生じたのを契機に当該条件が評価され、その評価に応じた分岐先の式を実行することを特徴とする。

【0020】また、前記部品編集機能は、前記部品間を接続するためのコネクタを有し、前記コネクタは、接続した部品が有する関係式に係る変数同士を論理的に結合し、その値の受け渡しを行うことを特徴とする。

【0021】また、接続した部品は、外部プログラムによって作成された部品であることを特徴とする。

【0022】尚、本発明におけるマルチメディアコンテンツとは、例えば、研究発表や製品デモなどのプレゼンテーション、教育用ソフト、エデュテインメント・ソフト (Edutainment Software)、ゲーム、百科事典、マルチメディアタイトル、製品カタログ、広告、仕様書、文書、スクリーンセーバー、アイコン、など、デジタル化された情報を全て含む。

【0023】また、本発明に係るオーサリングとは、テキスト、グラフィックス、サウンド、動画といった様々な形態のデータ(素材)を、1つのマルチメディア・コンテンツとしてまとめあげる作業のことをいう。

【0024】また、本発明に係るオーサリングプログラムを記録した記録媒体は、CD-ROMなどのパッケージ媒体や、インターネットなどの伝送媒体などを含む。

【0025】

【発明の実施の形態】以下、本発明の実施形態について図面を用いて説明する。

【0026】図1は本発明に係るマルチメディアオーサリングツールの一実施例を示す概略構成図である。

【0027】図1に示すように、このオーサリングツールは、CPU7やメモリ8を搭載した本体1、各種素材データやユーザからの指示内容を入力するマウス等のポインティングデバイスやキーボードなどからなる入力装置2、作成中のコンテンツや作成したコンテンツの実行結果を表示するディスプレイなどの表示装置4、プリンタやスピーカなどの出力装置5、サウンド(MIDI、WAVE等)機器などの拡張機器3、本発明に係るオーサリングプログラム9を保持する記憶装置6などから構成される。素材データとしては、例えば、動画、静止画、音声、図形、文字、記号、文章、プログラム等、デジタル素材となり得るデータが全て含まれる。

【0028】また、上記以外にも、静止画の入力手段としてデジタルカメラやスキャナ、動画の入出力手段としてデジタルビデオカメラ、音声の入力手段としてマイクロフォンなどを加えても良い。

【0029】また、図示していないが、記憶装置6には上記ハードウェア及びソフトウェアを管理するOSやデバイスドライバ等も記憶されている。

【0030】記憶装置6に記憶されているオーサリングプログラム9は、必要に応じてメモリ8にロードされ、CPU7によって実行される。このオーサリングプログラム9は、部品編集機能10、実行及び保存機能20、別プログラム組込み機能30等の機能を有しており、部品編集機能10には更に、子部品編集機能11、ページ編集機能12、サウンド部品、グラフ部品編集機能13、図形ペンによる作図機能14、アニメーション設定機能15、関係式設定機能16、コネクタの配置・接続機能17等を有している。尚、本実施例では、このオーサリングプログラム9をC言語に準拠したモデルを例として説明するが、プログラム言語はこれに限定されるものではない。

【0031】以下、上記各機能について、マルチメディアコンテンツの作成手順例をモデルに詳細に説明する。

【0032】(新規部品の作成) 本発明では、全ての素材データを「部品」として扱う。部品は自由に組み合わせることで1つのファイルにまとめることができるが、最小単位まで分解すると、次の種類に分類することができる。

【0033】(1) 画像部品: ペイント系の画像であり、ページごとに違う画像を表示することができる。

【0034】(2) 背景部品: 同じくペイント系の画像であるが、ページを増やしても画像は変化しない。

【0035】(3) 図形部品: ドロー系の図形であり、ページを増やしても形は変化しないが、色や模様などは変更することができる。

【0036】(4) ペン部品: グラフィックスの作成・編集に使用する。

【0037】(5) パッケージ部品: グラフィックスの作成・編集に使用する。

【0038】(6) テキスト部品: 文字を入力できる部品である。

【0039】(7) サウンド部品: サウンドファイルを取り込んだ部品である。

【0040】(8) MIDI部品: MIDI形式のファイルを取り込んだ部品である。

【0041】(9) 表・グラフ部品: 表またはグラフを描画するための部品である。

【0042】(10) アイコン部品: 部品ファイルにアイコンを設定する場合に使用する。

【0043】(11) 実行部品: 外部プログラムへのリンクを持たせた部品である。

【0044】本発明では、画像や図形、音声、テキスト

はもちろん、作成した文書やシミュレーションまでもがすべて「部品」であり、オブジェクト指向における「オブジェクト」と同様の概念である。部品は、データと動作（関係式）の両方を持つことのできる単位であって、例えば、サウンド部品は音声データを内部に持ち、ページ番号を切り換えたり、アニメーション設定をすることで、所望するタイミングで音声データを再生することができる。

【0045】また、読み込みや保存といった作業も部品を単位として行われる。部品は、次に説明する「子部品」や「孫部品」を持つことにより階層構造を持たせることができ、データを作成する場合には、最初に種類を決めた新規部品を（親部品として）作成し、さらに子部品をその内部に配置していくことで完成させていく。

【0046】（子部品の配置）2つの部品を組み合わせると、そこには必ず親子関係が生じる。この親子関係とは、部品の階層構造（ツリー構造）のことで、本発明では、ある部品（親部品）の内部に他の部品（子部品）が内蔵されている状態を、「子は親に含まれる」という。

【0047】例えば、図2（a）に示す「カメ」の部品では、下段のカメ41が親であるとする、中段のカメ42a及びカメ42bは子、上段のカメ42は孫になる。カメ42bから見るとカメ43は子に当たり、カメ42a及びカメ42b同士は並列の関係である。尚、図2（b）は、同図（a）を真上から見たときの簡略図である。

【0048】カメ41が歩くと全体が移動し、右のカメ42bが動くときカメ43も動く。逆に、カメ43がカメ42bの上で回転したとしても、一番下のカメ41には何の作用も及ばない。つまり、親部品を移動した場合、子部品も親部品の一部として移動するが、逆に子部品のみを移動しても、親部品やその他の子部品は移動しないのである。これは移動動作だけでなく、拡大・縮小・回転などの各動作についても同様である。

【0049】子部品として扱うことができるデータの種類としては、画像だけでなく、テキスト、グラフ、表、サウンド、音楽、動画、実行形式のファイルなどである。

【0050】実際に子部品を配置する手順を例に説明する。

【0051】まず、部品パレットに用意された部品を選んで、配置したい場所の2点の範囲を指定して配置する。ここで指定した範囲のサイズに合わせて、部品のサイズが自動的に調整される。このように、位置だけでなく範囲を定めるようにすることで、配置する場所とそのサイズを同時に設定することができ、配置後に更にサイズを調節する手間を省くことができる。また、外部から部品を読み込んで配置する場合には、配置したい場所を1点だけ指定することで、元の部品のサイズを優先させるようにしている。

【0052】例えば、図3（a）に示すように、メータ44の子部品として、針45の画像部品と「mA」46のテキスト部品を配置すると、図3（b）に示すような親子関係を持ったメータ部品47aができあがる。このメータ47bを利用して、子部品「mA」を「kV」に置きかえることで、図3（c）に示すような別のメータ部品47bを作成することもできる。

【0053】また、本発明では、子部品に対して親部品の座標系をそのまま適用するようにしているため、図3（d）に示すように、親部品のメータ47cを変形（拡大・縮小・回転など）すると、その変形に合わせて針や「mA」といった子部品も変形する。通常、このような変形動作に伴う座標計算は、階層が深くなるとそれぞれの部品の座標計算を反復して行う必要があるが、本発明では、座標系の変換が一次式になることに着目し、4つの基準点の写像位置を元にした連立方程式を解くことにより求めた係数を使用することで実現している、つまり、基準点を元にして求めた係数を利用して一次式の座標計算として計算することにより、階層の深さによる影響を受けにくいようにしているのである。

【0054】この座標系の変換方式によるオーサリングプログラム9の処理手順を図4に例示する。本発明による座標系の変換方式は、複数階層下の部品を最上位の座標へ投影する方式であり、階層分だけ座標変換を繰り返した後に求めた4点の座標を使用して、ガウスの除去法で写像の一次式の係数を求めて、イメージを高速に写像する方法である。

【0055】図4において、まず、初期処理として、現在の階層Lとその階層Lにおける部品の4隅の座標X[4]、座標Y[4]を設定する（Step01）。

【0056】次に、現在の部品上での座標を1つ上の階層へ写像するために、4つの座標全てに対して座標の計算を行う。

【0057】

【数1】

$$dx = X \times \text{部品の横拡大率}$$

$$dy = Y \times \text{部品の縦拡大率}$$

$$\cos D = \cos(\text{部品の回転角度})$$

$$\sin D = \sin(\text{部品の回転角度})$$

$$x = dx \times \cos D + dy \times \sin D + \text{部品のX座標}$$

$$y = dx \times \sin D + dy \times \cos D + \text{部品のY座標}$$

この座標変換の計算を現在の階層から最上位の階層まで繰り返し実行する（Step02～Step04）。

【0058】次に、上で求めた4座標の写像の式を8個作成し、以下に示すようにガウスの除去法を用いて係数Confを求める（Step05）。

【0059】

【数2】

$$\text{Conf}[0] \times X + \text{Conf}[1] \times Y + \text{Conf}[2] = \text{SrcX}$$

$$\text{Conf}[3] \times X + \text{Conf}[4] \times Y + \text{Conf}[5] = \text{SrcY}$$

式2に示した2つの式が4隅のXYに対して存在するので、実際には式は8個となる。以下に示す関数Solve()を用いて、この6変数、8式を解いて係数Confを求め

る。

【0060】

【数3】

SrcX[4] = 写像前のX座標

$$\text{SrcX} = \text{Conf}[0] \times \text{SX} + \text{Conf}[1] \times \text{SY} + \text{Conf}[2]$$

$$\text{SrcY} = \text{Conf}[3] \times \text{SX} + \text{Conf}[4] \times \text{SY} + \text{Conf}[5]$$

$$\text{TopImage}[Y][X] = \text{Image}[\text{SrcY}][\text{SrcX}]$$

尚、式4において、Image[XY座標]は写像前の部品のイメージデータであり、TopImage[XY座標]は写像先のイメージデータである。

【0062】以上のようにして、階層が深くなった場合でも、イメージを高速に写像することができるのである。

【0063】続いて、既に配置済みの部品に対する部品の置き換え操作について説明する。

【0064】図5に示すように、既に配置済みの部品を選択すると、選択された部品は点滅する囲み線49で囲まれた状態になる。この囲み線49で囲まれた部分に対して新たな部品を配置することで、部品の置き換えを行うことができる。

【0065】その場合、置き換え操作の種類を選択するためのダイアログボックス50が表示され、その中から以下の操作を選択する。

【0066】(1) 子部品を含む部品全体を置き換える操作

(2) 置き換える部品の子部品はそのまま、部品の画像のみを置き換える操作

(3) 配置する部品を次のページの画像としてページを追加する操作

部品の置き換えの場合、元の部品の位置、角度、サイズなどの属性が継承される。画像として置き換える場合には、子部品については元の部品のままとする。

【0067】続いて、部品のロック機能について説明する。

【0068】部品サイズの変更を禁止するための機能として部品のロック機能がある。この機能を使えば、子部品の親部品から見た目の拡大・縮小率を固定して、親部品の変形による変化を無効することができる。つまり、ロックされた部品は、親部品を縦横に拡大・縮小しても、見た目の大きさは変化しない。

【0069】この機能を利用すると、例えば、図6に示すような矢印の部品などで、矢印全体を拡大・縮小しても矢の部分のサイズが変わらないようにすることができる。図6(a)において、(1)の矢印部品は拡大・縮小前の部品、(2)の矢印部品はロックなしでの拡大を行った部品、(3)の矢印部品はロックした状態で拡大を行った部品を示している。

SrcY[4] = 写像前のY座標

Solve(X, Y, SrcX, SrcY, Conf)

次に、上で求めた係数Confを用いて画像データの描画を行う(Step06)。尚、SYは写像先の縦方向範囲、SXは写像先の横方向範囲の繰り返しである。

【0061】

【数4】

【0070】ロック機能を設定するには、矢印の柄の部分と矢の部分とを別々の部品で作成し、柄の部品を親部品とし、矢の部品を子部品として構成し、矢の部品をロック状態に設定する。このように設定して、親部品である柄の部品を横方向に引き伸ばしたとしても、図6(c)に示すようにロック状ある矢の部品52は、親部品の拡大率を打ち消す形で横方向に縮小される。これにより、結果として矢の部品は拡大前のサイズと同サイズで表示されることになる。ロックしない場合には、親部品である柄の部品を横方向に引き伸ばすと、図6(b)に示すように矢の部品51も親部品の拡大率に準じて拡大される。

【0071】このロック機能に関するオーサリングプログラム9の処理動作の流れを図7に例示する。図7の流れ図において、MagLockFは拡大・縮小ロックフラグ、FmagXは部品のX方向拡大率、FAngleは部品の回転角度、Parent->FmagXは親部品のX方向拡大率、Parent->FmagYは親部品のY方向拡大率を示す。

【0072】はじめに、当該部品がロック状態であるかどうか判定し(Step11)、ロックされていない場合には、通常の拡大率、つまり親部品の拡大率に準じた拡大率FmagXを返す(Step14)。

【0073】一方、当該部品がロック状態である場合には、更に、部品の回転角度を判定し(Step12)、回転角度が0度の場合には、親部品の拡大率を打ち消す拡大率FmagX/Parent->FmagXを返す。

【0074】回転角度が0度でない場合には、以下のよう

に、回転させた拡大率を使用して親部品の拡大率を打ち消すための拡大率FmagX/Magを返す。

【0075】

【数5】但し、 $\text{Mag} = \text{hypot}(\cos(\text{FAngle}) \times \text{fabs}(\text{Parent} \rightarrow \text{FmagX}), \sin(\text{FAngle}) \times \text{fabs}(\text{Parent} \rightarrow \text{FmagY}))$; であり、hypotは、直角三角形の直角をはさむ2辺の長さから斜辺の長さを計算する関数である。

【0076】(ページ編集機能) 部品の構造には、先に説明した親子関係による構造以外に、「ページ」の概念による構造がある。親子関係による構造は階層構造に基づく部品同士の関係を意味しているが、「ページ」の概念による構造の場合は1つの部品内部における並列的構造である。

【0077】部品は、複数のページを持つことができ、このページ機能をアニメーションなどの動画だけでなく、シミュレーションを行う上での基本的な要素として利用することができる。例えば、ON/OFFの2つの状態を持つスイッチのような部品では、0ページ目（ページ番号：0）をOFFの状態、1ページ目（ページ番号：1）をONの状態にすることができ、このページ番号を切替えることでスイッチの画像の表示を切替えるだけでなく、後に説明する関係式を結びつけることで、スイッチの状態に応じたシミュレーションモデルの状態を変移させることができる。

【0078】また、ページ番号は小数点以下の値を含むように設定することもでき、例えば図8に示すように、0ページ目に角度-4.5度の針を子部品として配置し（同図（a））、1ページ目に角度0度の針（図示せず）、2ページ目に角度+4.5度の針を配置することで（同図（c））、0.7ページ目に-13.5度の針を持たせることができる（同図（b））。このように、ページ間における子部品の位置や角度などの属性は、ページ番号の少数部分の値の比率で線形補間され、中間の状態を表現することができる。

【0079】このページ補間機能は、アニメーションにも適用することができる。従来、コマ送り方式のアニメーションを作成しようとする、非常に多くのページが必要となる。仮に、10コマ/秒で1分間のアニメーションを作成しようとする、600ページ必要となる。ところが、上記ページ補間機能を利用すると、図9に示すように、2ページで作成することができる。まず、同図（a）に示すように、1ページ目の背景53に対して子部品である車54を始点位置に配置する。次に、このページを複製して2ページ目を作成し、2ページ目の車54の位置を終点位置に配置する（同図（b））。これだけの作業で、車54が始点位置から終点位置へ移動するアニメーションが完成する。始点位置から終点位置までの途中のコマは、実行時にページ補間機能が自動的に補間する。

【0080】また、コマ送り方式によるアニメーションを作成する場合でも、図10に示すように、背景53

（同図（a））と車54の画像（同図（b））を別々に用意しておいて、背景53のページに車54の部品を配置し、背景54のページを追加しながら各ページの車53を少しずつ動かしていく。出来上がったページの表示を連続で切替えると、車54が移動するアニメーションが出来上がる（同図（c））。この場合、ページ毎に配置されている車54の部品は、全て同じ部品である。つまり、ページにまたがって配置した部品は、1つの部品内で同じ部品として扱われるため、ページを追加する度に画像のサイズが増えるといった問題を解決することができる。

【0081】（サウンド部品、グラフ部品編集機能）子

部品として、サウンド部品や表、グラフ部品を追加することができる。サウンド部品のデータ形式としては、例えば、PCM（パルス符号変調）方式により、アナログ音声をデジタル化したWAV形式などがある。このような音声データは部品として読込んでウィンドウに配置することで、その音声を再生することができる。

【0082】サウンド部品を適当なページに配置すれば、そのページが表示された時点で音声の再生が始まるので、例えば、ナレーションや効果音として利用することができる。また、ページ内の配置位置を左右の音量バランスとして用いることで、例えば、1ページ目の配置位置を右に、2ページ目の配置位置を左にすれば、右から左へ音声が移動するという効果音を作成することもできる。この場合にも、先に説明したページ間の線形補間機能を用いることで、滑らかな音の移動を実現することができる。

【0083】また、配置するサウンド部品の回転角度を音声の再生速度として用いると、例えば、1ページ目は角度0度、2ページ目を角度180度というように配置することで、再生周波数が段々と変化する効果を得ることができる。これらの効果を、後で説明する関係式によってページを切替えることで、音に対しての表現をもたせることができる。

【0084】また、表やグラフ部品を配置して、任意の値の棒グラフや円グラフを表示することも可能であり、後に説明するコネクタをグラフに接続して変数を割当てすることで、シミュレーションにおける変数の値をグラフとして表現することができる。

【0085】（図形ペンによる作図機能）パレットから「図形ペン」を道具として選択し、通常のペンと同様にポイントをクリックして線を描くことで、図形部品を作成することができる。作成された図形は、作図に使用された図形ペンを子部品として内部に保持する。ペンも部品の一種でペンの内部に子部品として画像を持つことができ、子部品としての画像を入れかえることにより、例えば、「線路を描くペン」「星模様を描くペン」「木や森を描くペン」など、様々なバリエーションが可能となる。

【0086】また、ペンの子部品の画像にアニメーションを設定することで、作成した線に対してアニメーション効果を持たせることもできる。例えば、ペンの子部品として矢印の画像を内部に持たせ、その矢印画像にアニメーションを設定すれば、実行時に矢印が動き、方向を示すというような効果を持たせることができる。

【0087】また、「バケツ」部品を使って、特定の範囲や図形などに色や模様を面として描くことができる。この「バケツ」部品に対しても、その子部品にアニメーション画像を設定することで動きのある模様を付けることが可能である。

【0088】（アニメーションの設定）部品にアニメー

ションの設定を行うことで、部品のページを自動的に変移させるアニメーションを作成することができる。図11に示すように、アニメーション設定機能には、「基本設定」「マウス操作」「詳細設定」の3種類の設定ウィンドウがある。

【0089】図11(a)にアニメーションの設定における「基本設定」ウィンドウを例示する。この基本設定では、アニメーションの開始タイミングの設定として、表示開始時58と一定時間ごとの繰り返し実行59をチェックできるようになっている。表示開始時58にチェックを入れると、実行後すぐにアニメーションを開始する。一定時間ごとに繰り返し実行59にチェックした場合には、繰り返しの時間の間隔60を指定することができる。例えば、間隔をゼロにすると最終ページ到達後も連続して再実行される。また、間隔60を最小値3秒～最大値6秒と設定した場合には、3秒から6秒のランダムな間隔でアニメーションが開始されるようになる。

【0090】また、ページ単位にページごとの時間61を設定できるようになっており、例えば少ないページ数で長い時間のアニメーションを作成することも可能である。また、合計時間62を変更することもできるため、目的に合わせて全体の時間を調整することができる。

【0091】アニメーション時には、上記のように指定した時間で自動的にページ番号が変移していき、画像が複数ある場合は、画像が切り替わる。また、子部品を持った部品のページ番号の変化は、先に説明したように子部品の位置をページ間で補間して移動や回転といった動作を実現する。

【0092】アニメーションは単に表示を切り替えるだけでなく、ページ番号の変化を発生させ、部品の状態を変化させる。ページ番号を含む関係式を内蔵している場合には、式の再評価が行われるので、指定した時間でシミュレーションのモデルを変更することが可能である。

【0093】例えば、10秒おきに2秒だけゲートが開くモデルを作成する場合などは、アニメーション機能で0ページ目の時間を8秒にし、2ページ目の時間を2秒に設定し、一定時間ごとの繰り返し実行の設定を行えば良い。

【0094】また、部品に、後に説明する関係式、例えば「GateOn = (@PageNo < 1);」を持たせることによって、実行時にこのGateOn変数を変化させ、後に説明するコネクタを使って伝播することも可能である。

【0095】また、例えば、関係式「OutData = sin(@PageNo * 3.141593);」を使って、時間軸に沿った値発生機を作成することもできる。

【0096】すなわち、ページ番号を時間軸に沿って変化するアニメーション機能は、見た目での変化を発生させるだけでなく、状態の変化としてシミュレーションモデル内で機能させることを目的としている。このようなページ番号の変化を利用するためにオーサリングプロ

グラム9内では、アニメーション設定が行われている場合にはCPU速度に無関係に、例えば10msごとにページ番号の再評価を行っている。ページ番号の変化が一定時間ごとでない場合には、表示とは無関係に内部では一定時間で変化させている。実際の画面上での表示の更新はCPU速度に合わせて間引きした形で行われる。

【0097】このような実際のアニメーションと関係式の動作に関して、アニメーション実行時におけるオーサリングプログラム9の処理の流れを図12に例示する。

【0098】図12の流れ図において、はじめに、アニメーションが設定されているかどうかチェックする(Step21)。アニメーションが設定されていない場合には、表示の更新処理を行う(Step26)。

【0099】一方、アニメーションが設定されている場合、経過時間d(ミリ秒)を次式により求める(Step22)。

【0100】

【数6】 $d = \text{現在時刻} - \text{最終演算時刻}$

経過時間dが10ミリ秒以上経過している場合は、以下の処理(Step24, 25)を繰り返し実行する(Step23)。但し、10ミリ秒未満になったら、表示の更新処理を行う(Step26)。

【0101】まず、次式により10ミリ間隔でページ番号PageNoを更新する(Step24)。ここで、AnimTimeは各ページ毎のアニメーション時間61である。

【0102】

【数7】 $\text{PageNo} = \text{PageNo} + 10 \cdot \text{AnimTime}$

経過時間d(残りの実行時間)と最終演算時刻を更新する(Step25)。

【0103】

【数8】 $d = d - 10$

最終演算時刻 = 最終演算時刻 + 10

続いて、アニメーションの設定における「マウス操作」の設定機能について説明する。

【0104】図11(b)にマウス操作設定ウィンドウを例示する。このマウス操作設定ウィンドウでは部品にマウス操作の設定をすることができ、シミュレーション実行時にマウスで操作を行うことができる。つまり、アニメーションやシミュレーションを実行中の部品に対して、クリックやドラッグ等のマウス操作を行った場合、その部品自身の操作について設定するのである。

【0105】マウス操作の設定には、大きく分けると、部品に対する操作の設定63とその部品の親部品に対する操作の設定64の2種類の設定がある。部品に対する操作の設定63は、以下の種類がある。

【0106】(1) 部品上でマウスボタンを押すことでページ番号を増減させる操作

(2) アニメーションの実行開始/停止を切り替える操作

(3) ドラッグで部品を移動させる操作

最も単純な操作としては、左ボタンで次ページ、右ボタンで前ページに遷移させる操作である。この設定を行うと、部品の上でボタンを押すたびに次ページへあるいは前ページへと切り替えることができる。このページの切り替え操作は、通常の場合は整数のページ番号により前後のページに切り替わるが、後に説明する「詳細設定」にてアニメーション効果の時間を調節することで、小数点以下の数値を含んだページ番号により滑らかに変化させることが可能である。

【0107】親部品に対する操作の設定64は、子部品に対してマウス操作したときに、対応する親部品のページに対する命令の設定を行うものである。命令としては、以下の通りの内容である。

【0108】(1) マウス・クリックによる指定ページへの移動

(2) アニメーションの開始/停止の指示

(3) ドラッグで移動や回転をさせることで親部品のページ番号の変更

親部品のアニメーションを開始/停止する操作の用途としては、例えば、親部品であるモーターの子部品としてスイッチを配置して、スイッチをクリックすることでモーターの動作を開始/停止するようなアニメーションやシミュレーションなどに使用される。

【0109】子部品をドラッグで移動や回転をさせることで親部品のページ番号の変更する操作の用途としては、子部品として配置されたつまみをつかんで（マウスでドラッグして）移動や回転を行い、親部品のページ番号を変更するようなアニメーションやシミュレーションなどに使用される。この時、つまみの移動距離や回転角度に応じて補間されたページ番号が設定されるので、例えば、子部品を0ページ目に角度0度で配置し、1ページに角度180度で配置しておき、実行時に回転操作をした子部品の状態がページ間のどの位置に相当するかを調べ、該当するページ番号が設定される。例えば、マウ

```
ChildData = Parent->PageStrList->Object[(int)Parent->PageNo]
X1 = ChildData->PosX
Y1 = ChildData->PosY
ChildData = Parent->PageStrList->Object[(int)Parent->PageNo+1]
X2 = ChildData->PosX
Y2 = ChildData->PosY
```

次に、X方向とY方向のどちら側の移動量が多いかを判定する (Step34)。

【0115】判定した結果、X方向の移動量の方が多い場合には、X方向における移動比率とその比率に応じた親部品のページ番号を求め、求めた親部品のページ番号に更新する (Step35)。

【0116】

【数11】 $d = (X - X1) / (X2 - X1)$
 $Parent->PageNo = (int)Parent->PageNo + d$
判定した結果、Y方向の移動量の方が多い場合には、Y

スのドラッグ操作で子部品を90度回転した場合は、ページ番号が「0.5」に設定される。このように、親部品のページ番号の値を変数として、子部品のドラッグ操作に連動して変数の値を変化させることにより、操作に応じた滑らかな動作をする部品を作成することができる。これにより、例えば、回転するつまみを持つ抵抗部品やボリュームなどを簡単に作成することができる。

【0110】子部品をドラッグで移動させることで親部品のページ番号の変更する機能に関して、オーサリングプログラム9の処理動作の流れを図13に例示する。この処理は、アニメーションやシミュレーション実行中に、マウス移動イベントの中で呼ばれる処理である。

【0111】図13の流れ図において、まず、マウスによって子部品がドラッグ中かどうかを判定する (Step31)。ドラッグ中でない場合、つまりドラッグ操作をされない場合には、本機能は無関係である。

【0112】ドラッグ中である場合には、次に、マウスによってドラッグされた子部品の位置座標を取得して、その移動量X、Yを計算する (Step32)。

【0113】

【数9】

MX = マウスのX座標

MY = マウスのY座標

$X = MX - \text{前回のマウスX座標} + \text{現在の部品のX座標}$

$Y = MY - \text{前回のマウスY座標} + \text{現在の部品のY座標}$

前回のマウスX座標 = MX

前回のマウスY座標 = MY

次に、前ページにおける子部品の位置X1、Y1と、次ページにおける子部品の位置X2、Y2を取得する (Step33)。

【0114】

【数10】

```
ChildData = Parent->PageStrList->Object[(int)Parent->PageNo]
X1 = ChildData->PosX
Y1 = ChildData->PosY
ChildData = Parent->PageStrList->Object[(int)Parent->PageNo+1]
X2 = ChildData->PosX
Y2 = ChildData->PosY
```

方向における移動比率とその比率に応じた親部品のページ番号を求め、求めた親部品のページ番号に更新する (Step36)。

【0117】

【数12】 $d = (Y - Y1) / (Y2 - Y1)$

$Parent->PageNo = (int)Parent->PageNo + d$

続いて、図14は、子部品をドラッグで回転させることで親部品のページ番号の変更する機能に関して、オーサリングプログラム9の処理動作の流れを例示した図である。この処理も、アニメーションやシミュレーション実

行中に、マウス移動イベントの中で呼ばれる処理である。

【0118】図14の流れ図において、まず、マウスによって子部品がドラッグ中かどうかを判定する (Step41)。

【0119】ドラッグ中である場合には、次に、マウスによってドラッグされた子部品の位置座標を取得して、その回転角度Aを計算する (Step42)。

【0120】

【数13】

MX = マウスのX座標

ChildData = Parent->PageStrList->Object[(int)Parent->PageNo]

A1 = ChildData->Angle

ChildData = Parent->PageStrList->Object[(int)Parent->PageNo+1]

A2 = ChildData->Angle

次に、回転角度の比率とその比率に応じた親部品のページ番号を求め、求めた親部品のページ番号に更新する (Step44)。

【0122】

【数15】 $d = (A - A1) / (A2 - A1)$

Parent->PageNo = (int)Parent->PageNo + d

また、図11(b)に示したマウス操作設定ウィンドウにおいて、マウス通過時に次ページを表示65にチェックを入れると、ポインタが部品と重なっている間、その部品の次のページを表示する。この機能を使えば、ホットスポットとした部品を作成することができる。

【0123】続いて、アニメーションの設定における「詳細設定」機能について説明する。

【0124】図11(c)に詳細設定設定ウィンドウを例示する。この詳細設定設定ウィンドウでは、ページの表示順66とアニメーション効果67に関する設定を行う。ページの表示順66には次の選択肢がある。

【0125】(1) 繰返し動作

(2) 往復動作

(3) 最終ページまで動作

また、コマ送りによる表示機能や、アニメーション停止時に自動的に先頭ページへ戻す機能なども選択することができる。

【0126】アニメーション効果67では、ページ間の画像の変化、例えば、スクロール、ワイプ、ディゾルブなどを指定する。

【0127】また、アニメーション効果の時間を設定することで、ページ番号の変化に対しての効果の時間を設定することができる。例えば、0ページから1ページまでを4秒かけて変化させる基本設定のアニメーションに対して、アニメーション効果時間を1秒に指定した場合は、アニメーション効果は3秒後の0.75ページから1ページまでの間だけ有効になる。この機能は、例えば、3秒間ページを表示した後に1秒間かけて次のページにスクロールさせるというように、ページを指定した時間だ

MY = マウスのY座標

DA = $\text{Atan}((MX - \text{前回のマウスX座標}) / (MY - \text{前回のマウスY座標}))$

前回のマウスX座標 = MX

前回のマウスY座標 = MY

A = 現在の部品の角度 + DA

次に、前ページにおける子部品の角度A1と、次ページにおける子部品の角度A2を取得する (Step43)。

【0121】

【数14】

け停止してからスクロールして、次のページへ移動するというような動作を実現したい場合などに使用することができる。このアニメーション効果37の時間が設定されている場合は、マウス操作でのページ番号変更時にも途中のページ番号が使用される。従って、目的ページへの変移は、指定した効果時間のアニメーションを伴って変化する。

【0128】(関係式の設定) 部品には、それぞれ関係式を設定することができ、設定した関係式に変数と変数の有機的関係を記述することで、部品単位のシミュレーションモデルを作成することができる。また、部品間における変数のリンクは、後に説明するコネクタによって接続することにより行う。

【0129】例えば、式「 $A = C + 2$ 」を通常のプログラムとして実行する場合を考えると、 $C + 2$ の計算が実行され、その結果が変数Aに代入される。シミュレーションのモデルとして、この式「 $A = C + 2$ 」を実行する場合に、どのタイミングで実行するのかという問題が発生する。一番単純な実行タイミングは、シミュレーションの最小時間間隔で一定時間ごとに実行する方法であるが、モデルが複雑になるに従って無駄な計算が多くなり全体としての処理速度を低下させることになる。つまり、仮にCが一定の値であって変化しない状態であっても、常にAへの代入処理が定期的に繰り返されるために、処理時間が累積して行くことになる。

【0130】本発明に係る関係式においては、式を代入式としてでなく、関係を表す式として扱うことで、無駄な計算時間の問題を解決すると共に単純な記述を可能にする。つまり、式「 $A = C + 2$ 」は、 $C + 2$ の結果が変数Aの値と等しくなる関係を示しているとの考え方を基本とする。Aの値は、Cの値が変化した場合のみ変化するの、上式はCの値が変化した場合にのみ実行すれば良い。

【0131】同様に、Cに値を代入する別の関係式に対しては、その関係式が参照している変数に関しての、更

新要求が発生した場合のみ評価を行う。

【0132】そのために、オーサリングプログラム9内部では、全ての関係式と変数のリンク（参照テーブル）を作成し、どの変数がどの式に影響を及ぼすかを動的に更新する。これにより、単純な記述による関係式で無駄な計算時間の問題をも解決することができる。

【0133】以上説明した関係式の機能を実現するための変数の参照テーブルを図15に例示する。関係式の構文解析と変数の参照テーブルの作成は、例えば、コンパイラがプログラムをコンパイルする際に行う方式と同様の方式で行う。図15(a)の式1から同図(b)に示すような参照テーブルが作成され、変数番号0が変数Aの情報、変数番号1が変数Bの情報である。これにより、変数Bの値に変化が生じた場合には、ポインタRefListが指し示す式1を評価する処理を行うことができる。

【0134】この参照テーブルを用いたオーサリングプログラム9の処理例を図16に示す。

【0135】この関係式に係る処理は、ある変数の値に変化が生じたとき（変数の値が更新されたとき）に、再帰で呼出される。

【0136】まず、更新が発生した変数の変数番号をもとに参照テーブルから当該変数を参照している式のリストを取得し（Step51）、取得したリストの数だけ、以下の処理（Step53～Step58）を繰り返す（Step52）。

【0137】次に、式のリストから式へのポインタを取得し（Step53）、ポインタが指し示す先の式が使用可能な状態であるかどうかチェックする（Step54）。

【0138】使用可能状態である場合には、当該式を使用中の状態に設定し（Step55）、当該コードを実行コードを実行する（Step56）。

【0139】更に、当該式の代入先の変数に対して、本処理を再帰で呼出す（Step57）。呼出す際の引数は代入先の変数番号である。

【0140】当該式を使用可能状態に戻す（Step58）。

【0141】続いて、関係式内の方程式について説明する。

【0142】本発明に係る関係式においては、関係式の一部あるいは全部に方程式を記述することができる。扱うことができる方程式は、連立一次方程式である。

【0143】シミュレーションによっては、例えば、複数の電池と電球で構成されるモデルなどでは、電流と電圧の値を求めるために方程式を解く必要が生じる場合がある。本発明に係る関係式においては、例えば「 $A + 2 * B = 3$;」のように左辺に式を書くことで方程式を表現し、関係式の行に追加することで記載する。

【0144】上記方程式は、通常の関係式と同様に「AにBの2倍を加算した結果が3である」という関係を表している。ただし、通常の関係式が、変数のリンクをたどって順番に式の評価が行われるのに対して、方程式の

場合は、変数のリンクをたどって関連する方程式全てを一度に計算するという特徴がある。例えば、第1の部品に「 $A + C = 4$ 」、第2の部品に「 $C - B = 4$ 」、第3の部品に「 $A - B = 4$ 」が記述されているとして、それぞれの方程式の変数A、B、Cがコネクタによってリンクされている場合には、上記3つの式を使った連立方程式を計算する。

【0145】オーサリングプログラム9内部では、部品に記述されている方程式の変数のリンクをたどって、関連する変数の式を使って連立一次方程式のグループを自動的に作成し、各方程式の係数の行列を作成して解を求め、変数に結果を格納する。部品同士の接続関係は、部品の移動やページ番号の変更などで動的に変化するため、方程式のグループも動的に変化し、関係式の変数リンクを使用して方程式のグループを動的に更新する。

【0146】また、本発明に係る関係式においては、方程式は、係数として関係式内の変数を使用することもできる。例えば、「 $A + d * B = 0$;」という方程式の場合には、dは係数として使用される。この場合、dの変数の値が方程式の係数となり、別の関係式でdの値が変更されたときに、変数更新用のリンクによりこの方程式は再計算され、従って変数A、Bも係数dの新しい値に合った値に更新されることになる。

【0147】また、上で説明したように、方程式は動的に変化することがあるため、場合によっては式の数が増える変数の数より少なくなることが発生する。その場合には、変数を強制的に0として方程式を解いて、その結果が正しければその解を使用する。変数を0として方程式を解いた結果に矛盾が発生した場合には、別の変数を0として再度結果の検証を行う。ここで、式の数求める変数の数より少なくなる方程式とは、基準となる値がない状態で方程式になっている場合などのことで、変数を0として解を求めることで有効な解が求まるケースが多く存在する。

【0148】具体的には、例えば、電池と電球の回路を計算するような方程式の場合に、電圧の値は相対的となり、絶対値の0Vの存在がない場合に解が求まらない。このことは、各端子の相対的な電圧の差と電流の量が正しければ、大地に対しての絶対電圧値が何Vであってもかまわないことを意味しており、この場合にどれかの電圧変数を強制的に0にして値を求めることにより有効な解が求まるのである。

【0149】また、本発明に係る関係式においては、方程式に「Sum(A) = 0 ;」のようにSUM関数を使用することで、変数AをSUM変数として扱う。このSUM変数Aをコネクタに割り当てることで、変数Aはコネクタに接続された変数の総和の式として扱われる。SUM変数Aのコネクタに部品Bの変数Xのコネクタと部品Cの変数Yのコネクタを接続した場合に、「Sum(A) = 0 ;」は「 $X + Y = 0$;」として解釈されて方

程式が生成される。

【0150】続いて、関係式内の条件文について説明する。

【0151】本発明に係る関係式には、条件文を記述することもできる。この条件文は以下の形式で記述する。

【0152】

【数16】if (条件式) {式1} else {式2}

または

if (条件式) {式1}

上記条件式では、条件式が真の場合には、式1が有効で、偽の場合には式2が有効になる。有効である方の式(式1もしくは式2)が参照している変数の値が変化した場合に、式の再評価が行われる。式が有効であるか無効であるかという状態の変化は、その式の再評価のための変数と関係式のリンク状態を更新することになる。

【0153】また、式として方程式を記述した場合は、式が方程式のグループに追加され、式が無効になった時点で削除される。方程式グループの式の数は条件文によっても動的に変化することになる。

【0154】関係式が条件文を含む場合には、条件の式の有効/無効が設定されることで、処理を切換えることができる。この処理例を図17に示す。

【0155】図17に示した処理は、例えば、以下ののような条件文を含む関係式を処理する場合を示している。

【0156】

【数17】

```
if (A>0){  
    C=B+3;          . . . . 式1  
}else{  
    C=2*B;          . . . . 式2  
}
```

つまり、変数Aが更新されるのを契機に条件文if (A>0)が評価され、その評価結果に基づいて処理を行うのである。尚、図17の中で、FuncList1は条件文が真のとき有効な式のリスト(式1に相当)、FuncList2は条件文が偽のとき有効な式のリスト(式2に相当)である。

【0157】まず、条件式(A>0)の真偽のチェックを行い(Step61)、真の場合には、真の場合の式(式1)を有効にし(Step62)、偽の場合の式(式2)を無効にして(Step63)、有効にした真の場合の式(式1)を実行する(Step64)。

【0158】一方、偽の場合には、偽の場合の式(式2)を有効にし(Step65)、真の場合の式(式1)を無効にして(Step66)、有効にした偽の場合の式(式2)を実行する(Step64)。

【0159】続いて、本発明に係る関係式の内部で使用するシステム変数に関して説明する。

【0160】関係式内部には、部品の位置やページ番号などをシステム変数として記述することができ、システム変数を使用すると、部品のサイズやページなどを関係

式で参照または変更することができる。システム変数には、例えば、次のような変数がある。

【0161】

- (1) @PageNo: ページ番号
- (2) @PosX: 水平座標
- (3) @PosY: 垂直座標
- (4) @DrawRate: 不透明度
- (5) @MagX: 横拡大率
- (6) @MagY: 縦拡大率
- (7) @Angle: 角度

尚、先頭の@が1つの場合は自分自身を対象とし、@を2つにすると、親部品が対象になる。

【0162】関係式の中での使用例を説明すると、例えば、「A=@PageNo;」と記述することにより、変数Aは部品のページ番号に關係付けられる。これはページ番号の値が必要に応じて変数Aに代入されることを意味する。ページ番号がアニメーションの実行やマウス操作等で変化した場合に、変数Aへの代入処理が発生する。

【0163】逆に、「@PageNo=A;」と記述すると、部品のページ番号は変数Aに關係付けられる。変数Aの値が関係式や方程式、コネクタ接続などによって変化した場合に、部品のページ番号が変数Aの新しい値に更新される。

【0164】これらの更新タイミングについても、変数のリンクを元にして更新の要否を判断しているため、値の更新要求がない場合は代入が行われない。従って、アニメーション設定により部品が自分自身でページ番号を変化させている場合にも、変数Aからの更新の要求が発生しない間は、部品のページ番号はアニメーション設定時間で変化していく。変数Aからの更新要求が発生すると、その更新内容で@PageNoが更新され、その後もアニメーションは継続される。

【0165】(コネクタの配置と接続) 本発明におけるコネクタとは、部品に追加する特殊なオブジェクトで、部品同士を接続したり、部品にデータを受け渡す役割を持つ。例えば、関係式の変数を別の部品とリンクさせることもできる。

【0166】部品に追加したコネクタは、図18に示すように、丸に十字の形で部品の上に表示され、マウスで移動したりやサイズを変更したりすることができる。また、部品には複数のコネクタを追加することができ、コネクタに関係式の変数を割り当てることができる。

【0167】具体的に説明すると、例えば、部品Aのコネクタに部品Aの関係式「C=X;」の変数Cを、部品Bのコネクタに部品Bの関係式「@PageNo=A;」の変数Aを割り当て、それぞれのコネクタを重ねて接続すると、「C=X;」の関係式の変数Cと「@PageNo=A」の関係式の変数Aへのリンクが設定される。つまり、部品Bの@PageNoへ、部品Aの変

数Cの代入が発生することになる。従って、変数Xの値が変化した場合に、変数Cと変数Aを介して@PageNoへ変数Xの値が伝播することになる。この機能を用いると、ある部品（この場合部品A）から他の部品（この場合部品B）のページを変更するという処理を実現することができる。

【0168】方程式の場合にも、同様にコネクタに変数を割り当てることで変数を共有することができ、例えば、部品Aのコネクタに方程式「 $X + 2 * Y = 0$;」の変数Xを、部品Bのコネクタに方程式「 $E - 3 * F = 0$;」の変数Eを割り当てて接続することで、変数X=変数Eの式が生成され、方程式の計算に使用される。また、特定の変数を割り当てていないコネクタを2つ以上持つ部品は、コネクタ間の変数を共有するための接続線として使用され、これは画面上での離れた位置のコネクタを接続状態にするために使用することができる。

【0169】また、コネクタは、変数の共有や伝播だけでなく、部品を配置する時の基準位置としても使用される。コネクタにロックの設定をすることで、コネクタの接続や位置を出来るだけ維持する形で部品の回転や拡大縮小が行われる。ロックされたコネクタを持つ接続線の部品と接続されたコネクタを持つ四角の部品を移動することで、接続線のコネクタは四角の部品のコネクタと同じ位置を保持するために回転や拡大縮小の変形が行われる。変形される部品のロックされた2つのコネクタが水平か垂直に近い場合は回転とコネクタの配置方向への拡大縮小が行われ、そうでない場合は縦横の2方向の拡大縮小が行われる。

【0170】図18（1）と同図（2）は、四角を移動した場合にロックされたコネクタによる線部品の回転と拡大縮小の例であり、同図（3）はコネクタを対角線上に配置した場合の例で四角を移動させることで線部品は2方向に拡大縮小される。

【0171】（実行、保存機能）作成したシミュレーション等は、実行ボタンを押すことで実行することができ、設定したアニメーション設定情報に従って、ページ番号が変化して表示が変移する。また、設定したマウス操作設定情報により、部品をクリックすると次のページへ移動したり、部品をドラッグするとページ番号が滑らかに変化したりすることができる。また、設定した関係式により作成された式と変数のリンクにより、部品内部の変数の値が必要に応じて更新され、また、システム変数への関連づけにより、変数の値の変化を画面上に表示されている部品の位置やページの状態で確認することができる。

【0172】作成したシミュレーションは、例えば、以下のファイル形式で保存することができる。

【0173】

- (1) 通常の部品
- (2) 静止面

- (3) 音声
- (4) 文書
- (5) 表計算
- (6) 実行ファイル
- (7) スクリーンセーバー
- (8) 動画ファイル

実行ファイル形式で保存したファイルは、単体で実行することができ、動画ファイルで保存したファイルは、例えば、ホームページなどで動画配信として使用することもできる。

【0174】（別プログラムの組み込み機能）別プログラムによる実行形式のファイルも、部品として組み込む機能を有している。その部品の実行を開始させたいページに配置することで、実行時に外部のプログラムを起動することができる。この場合、（1）元の実行を停止して動作させる設定と（2）並行して動作させる設定の2種類の中から選択する。元の実行を停止させる設定を使うことで、複数のプログラムを連続して動作させることができる。部品として組み込むことのできる実行形式のファイルは、本発明に係るオーサリングプログラム9にて作成したシミュレーション自体も、実行ファイル形式に変換して配置することができる。また、組み込んだ部品を介して、外部のプログラムとのメッセージによる関係式の変数へのアクセス機能を有しており、複数の外部プログラムと連携した動作を行うこともできる。また、外部プログラムからメッセージを受け取り、内部部品の変数へ値を設定することもできる。この場合も値の変更を受けた変数から必要に応じて別の関係式への更新が行われる。

【0175】また、外部プログラムから変数の参照メッセージを設定することで、変数が更新されるたびに外部プログラムへメッセージが送信される。その後、外部プログラムから参照終了メッセージが送られれば、以後のメッセージの送信は行われない。

【0176】このように、外部プログラムとの通信機能により作成したシミュレーションを、ユーザーインターフェイスに優れたシステムとして構築することができる。つまみやメーターを持った制御パネルをグラフィカルに作成して実行形式で保存し、実際の処理を行う別プログラムと通信機能を介して動作させることにより、ユーザーインターフェイス部分と処理動作部分とを独立に作成することができる。また、特殊な計算が必要なシミュレーションの場合に、特殊な計算の処理部分は外部プログラムとして別の言語で作成することも可能である。

【0177】以上説明したように、本発明によれば、高度な技術的知識や複雑な操作を必要とせずに、視認性の高い効果的なデジタルコンテンツを簡単に作成、実行、配布することができる。本発明の用途としては、プレゼンテーション、製品紹介、PR広告、説明書、カタログ

など、広報や営業分野での利用、仕様書、工程管理図、品質管理図、取扱説明書など、製造分野での利用、教材、研究発表など教育分野などでの利用が見込まれる。また、作成したコンテンツは実行ファイル形式で保存できるため、他のパソコンユーザへ配布したり、インターネットなどのネットワークを介して配布したりすることも可能である。

【0178】以上、本発明の実施形態について詳細に説明したが、本発明は本実施例に限定されず、本発明の主旨を逸脱しない範囲において、種々の改良や変更を成し得るであろう。

【0179】従って、本発明はこの開示から妥当な特許請求の範囲に係わる発明特定事項によってのみ限定されるものでなければならない。

【0180】

【発明の効果】本発明によれば、高度な技術的知識や複雑な操作を必要とせずに、視認性の高い効果的なデジタルコンテンツを以下のように簡単に作成、実行、配布することができる。

【0181】(1) 文字や画像、音声などの様々なデータを容易に扱うことができる。

【0182】(2) 親子関係を基本とし、複数の子や孫を内蔵することができるため、様々な形式のデータを容易に組み合わせるコンテンツを作成することができる。

【0183】(3) 1つの部品に複数のページを持たせることができ、更にページ間の自動補間機能を持たせているため、ページを利用したアニメーションやシミュレーション等を容易に作成することができる。

【0184】(4) コネクタや関係式を用いて、部品に対して新しい機能を容易に追加することができる。

【0185】(5) 外部プログラムと連携した動作を行うことも簡単にできる。

【図面の簡単な説明】

【図1】本発明に係るマルチメディア・オーサリングツールの一実施例を示す概略構成図。

【図2】部品における親子関係の概念を示すイメージ図。

【図3】親子関係を有する部品を組み合わせるコンテンツを作成する様子を示したイメージ図。

【図4】親部品の変形に連動した子部品の変形に係る座標変換の処理手順を示したフローチャート。

【図5】既に配置済みの部品に対する部品の置き換え操作に関するイメージ図。

【図6】部品のロック機能に関するイメージ図。

【図7】部品のロック機能の処理手順を示したフローチャート。

ャート。

【図8】ページ間の自動補間機能に関するイメージ図。

【図9】ページ間の自動補間機能をアニメーションに応用した例を示すイメージ図。

【図10】ページ機能を用いてコマ送り方式によるアニメーションを作成した例を示すイメージ図。

【図11】アニメーション設定機能に関するイメージ図。

【図12】アニメーションの実行と関係式の動作に関する処理手順を示したフローチャート。

【図13】マウス操作の設定において、ドラッグによる移動操作に連動して親部品のページ番号を変更する処理手順を示したフローチャート。

【図14】マウス操作の設定において、ドラッグによる回転操作に連動して親部品のページ番号を変更する処理手順を示したフローチャート。

【図15】変数の参照テーブルのレイアウト例を示したイメージ図。

【図16】設定された関係式に関する処理手順を示したフローチャート。

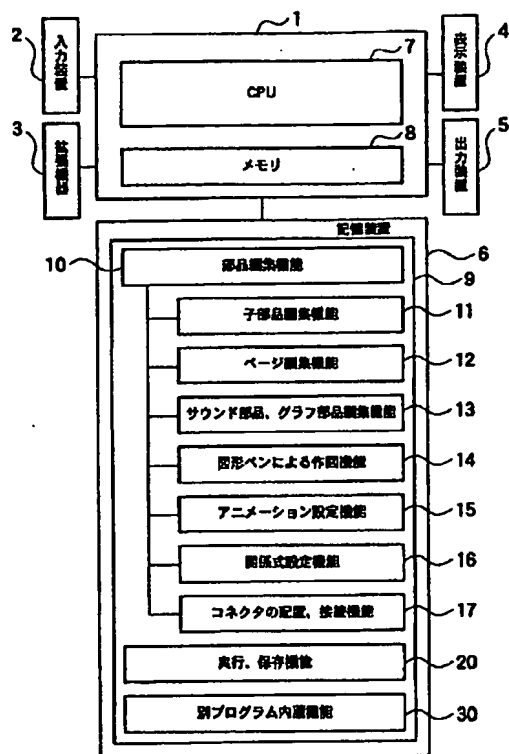
【図17】判定文を含む関係式に関する処理手順を示したフローチャート。

【図18】部品を接続するためのコネクタに関するイメージ図。

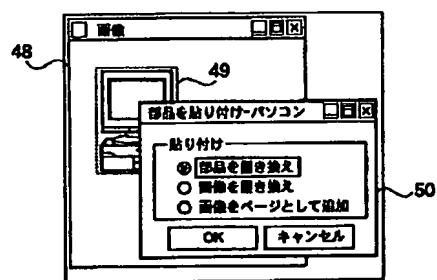
【符号の説明】

- 1... オーサリングツール本体
- 2... 入力装置
- 3... 拡張機器
- 4... 表示装置
- 5... 出力装置
- 6... 記憶装置
- 7... CPU
- 8... メモリ
- 9... オーサリングプログラム
- 10... 部品編集機能
- 11... 子部品編集機能
- 12... ページ編集機能
- 13... サウンド部品、グラフ部品編集機能
- 14... 図形ペンによる作図機能
- 15... アニメーションによる設定機能
- 16... 関係式設定機能
- 17... コネクタの配置、接続機能
- 20... 実行、保存機能
- 30... 別プログラム組込み機能

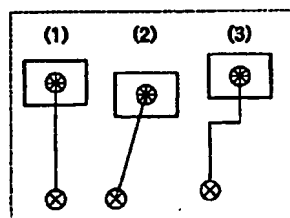
【図1】



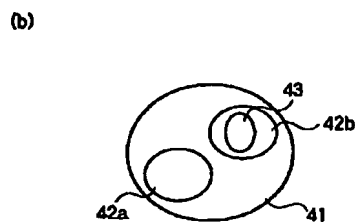
【図5】



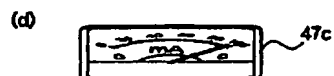
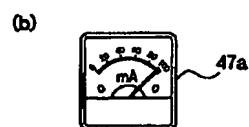
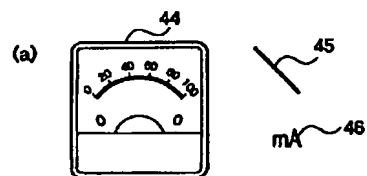
【図18】



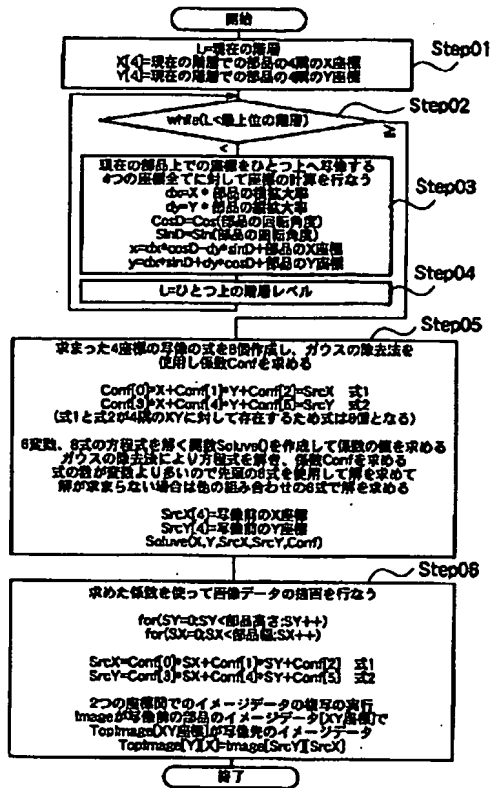
【図2】



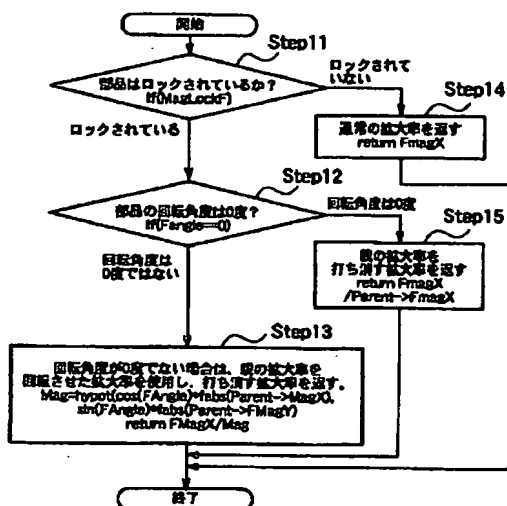
【図3】



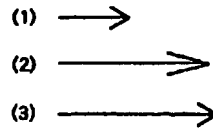
【図4】



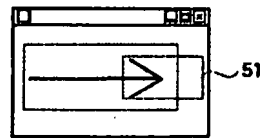
【図7】



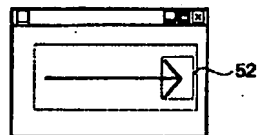
【図6】



(a)

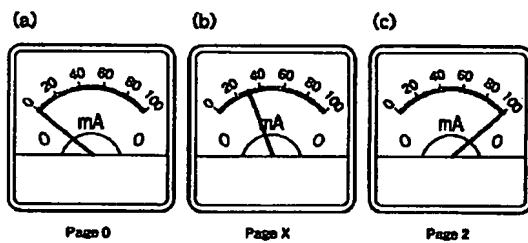


(b)

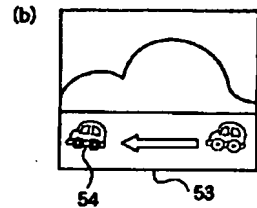
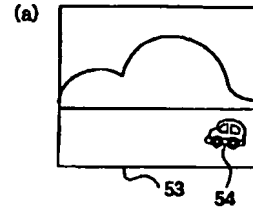


(c)

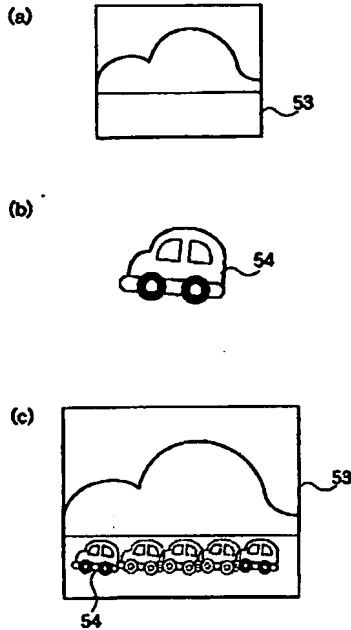
【図8】



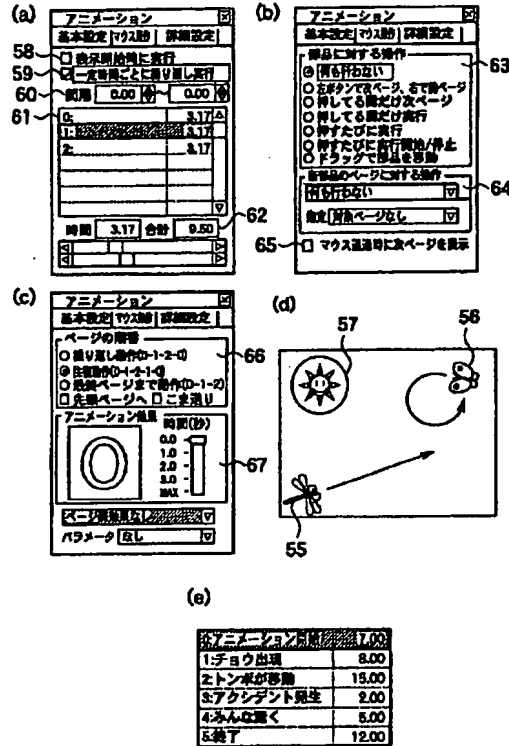
【図9】



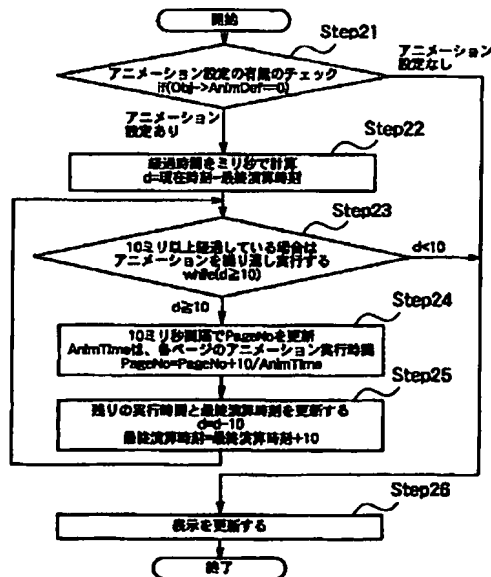
【図10】



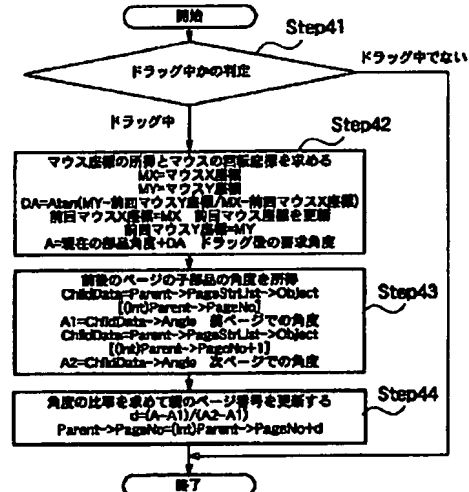
【図11】



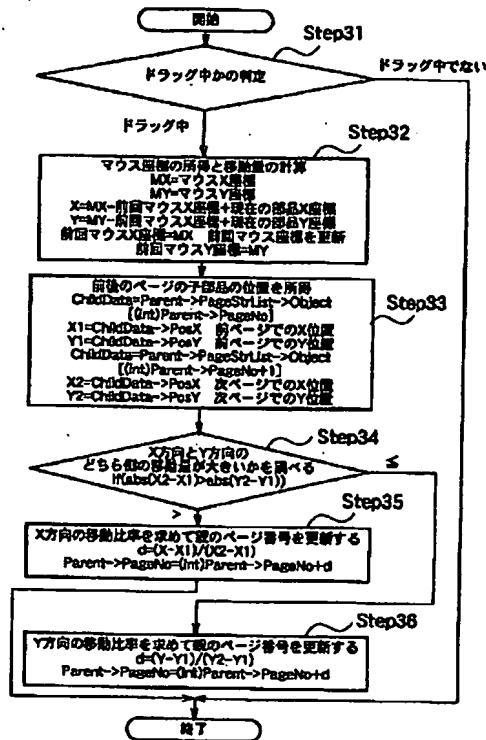
【図12】



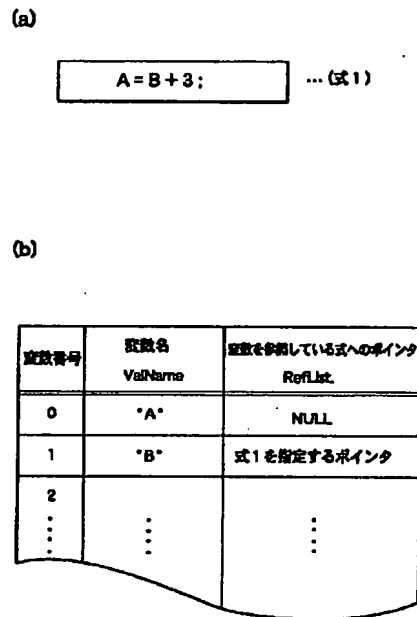
【図14】



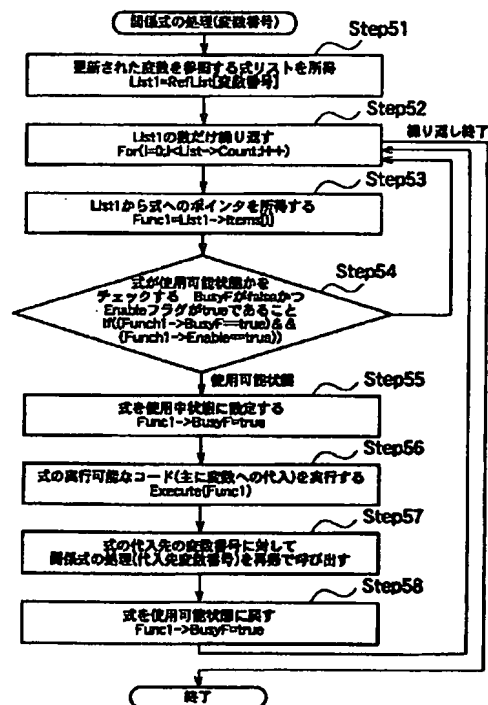
【図13】



【図15】



【図16】



【図17】

